

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A method for administration of network financial transaction terminals, comprising:

sending an event query to a management instrumentation application by a queued component client on one of the financial transaction terminals;

subscribing to a pre-determined event type by the queued component client;

receiving an event notification corresponding to said subscribed-to pre-determined event type from the management instrumentation application by the queued component client;

capturing and consuming an event message corresponding to said received event notification by the queued component client before writing the event message into an event log on said one of the financial transaction terminals;

sending the captured and consumed event message to a server site event queue by the queued component client via message queuing services components;

removing ~~the~~ said sent event message from the server site event queue by a queued component server; and

storing ~~the~~ said removed event message into a database by the queued component server.

2. (Original) The method of claim 1, wherein sending the event query further comprises sending a log event type of event query to the management instrumentation application.

3. (Original) The method of claim 2, wherein sending the log event type of event query further comprises subscribing to the log event type by the queued component client.
4. (Original) The method of claim 1, wherein receiving the event notification further comprises receiving log event type of event notification by the queued component client.
5. (Original) The method of claim 4, wherein receiving the event notification further comprises receiving the event notification by the queued component client acting as an event consumer.
6. (Canceled)
7. (Canceled)
8. (Previously presented) The method of claim 1, wherein receiving the event notification further comprises placing the log event message in a client site event queue by the queued component client.
9. (Original) The method of claim 8, wherein receiving the event notification further comprises creating the client site event queue by the queued component client.
10. (Original) The method of claim 4, wherein receiving the event notification further comprises receiving the event notification by the queued component client from the management instrumentation application when a log event occurs.
11. (Original) The method of claim 1, wherein sending the event message further comprises sending a log event message in extensible markup language to the server site event queue by the queued component client.

12. (Original) The method of claim 11, wherein sending the log event message further comprises placing the log event message in a client site event queue by the queued component client.
13. (Currently amended) The method of claim 12, wherein placing the log event message in the client site event queue further comprises creating the client site event queue by the queued component client.
14. (Original) The method of claim 11, wherein sending the log event message further comprises sending the log event message to the server site event queue over a network.
15. (Original) The method of claim 14, wherein sending the log event message over the network further comprises sending the log event message to the server site event queue over a proprietary network.
16. (Original) The method of claim 14, wherein sending the log event message over the network further comprises sending the log event message to the server site event queue over a public network.
17. (Original) The method of claim 1, wherein removing the event message further comprises removing a log event message from the server site event queue by the queued component server.
18. (Original) The method of claim 17, wherein removing the log event message further comprises removing the log event message in extensible markup language format from the server site event queue by the queued component server.
19. (Original) The method of claim 18, wherein removing the log event message further comprises removing the log event message from the server site event queue by the queued component server acting as an event processor.

20. (Original) The method of claim 1, wherein storing the event message further comprises storing a log event message into the database by the queued component server.
21. (Original) The method of claim 20, wherein storing the log event message further comprises storing the log event message in extensible markup language format into the database by the queued component server.
22. (Original) The method of claim 21, wherein storing the log event message further comprises storing the log event message into a structured query language server data warehouse by the queued component server.
23. (Original) The method of claim 22, wherein storing the log event message further comprises analyzing the stored log event message.
24. (Original) The method of claim 23, wherein analyzing the stored log event message further comprises analyzing the stored log event message using an online analytical processing application.
25. (Original) The method of claim 1, further comprising allowing a user to query the database via a web browser user interface.
26. (Original) The method of claim 25, wherein allowing the user to query the database further comprises filtering query results based on selections entered by the user on the user interface.
27. (Original) The method of claim 26, wherein filtering the query results further comprises displaying a report of the filtered results for the user via the user interface.
28. (Original) The method of claim 1, further comprising sending a notice of a security related event as an event notification to a predefined terminal for a system administrator.

29. (Original) The method of claim 28, wherein sending the notice of the security related event further comprises detecting the security event by a filtering mechanism associated with the database.

30. (Currently Amended) A system for administration of network financial transaction terminals, comprising:

a queued component client on one of the financial transaction terminals adapted for sending an event query to a management instrumentation application on said one of the financial transaction terminals, for subscribing to a pre-determined event type, for receiving an event notification corresponding to said subscribed-to pre-determined event type from the management instrumentation application, for capturing and consuming an event message corresponding to said received event notification before writing the event message into an event log on said one of the financial transaction terminals, and for sending the captured and consumed event message to a server site event queue via message queuing services components; and

a queued component server adapted for removing the said sent event message from the server site event queue, and for storing the said removed event message into a database.

31. (Previously presented) The system of claim 30, wherein the queued component client is further adapted for sending log event type of event query to the management instrumentation application.

32. (Previously presented) The system of claim 31, wherein the queued component client is further adapted for subscribing to the log event type.

33. (Previously presented) The system of claim 30, wherein the queued component client is further adapted for receiving a log event type of event notification.

34. (Previously presented) The system of claim 33, wherein the queued component client is further adapted for receiving the event notification acting as an event consumer.

35. (Canceled)

36. (Canceled)

37. (Previously presented) The system of claim 30, wherein the queued component client is further adapted for placing the log event message in a client site event queue.

38. (Previously presented) The system of claim 37, wherein the queued component client is further adapted for creating the client site event queue.

39. (Previously presented) The system of claim 33, wherein the queued component client is further adapted for receiving the event notification from the management instrumentation application when a log event occurs.

40. (Previously presented) The system of claim 30, wherein the queued component client is further adapted for sending a log event message in extensible markup language to the server site event queue.

41. (Previously presented) The system of claim 40, wherein the queued component client is further adapted for placing the log event message in a client site event queue.

42. (Previously presented) The system of claim 41, wherein the queued component client is further adapted for creating the client site event queue.

43. (Previously presented) The system of claim 40, wherein the queued component client is further adapted for sending the log event message to the server site event queue over a network.

44. (Previously presented) The system of claim 43, wherein the queued component client is further adapted for sending the log event message to the server site event queue over a proprietary network.
45. (Previously presented) The system of claim 43, wherein the queued component client is further adapted for sending the log event message to the server site event queue over a public network.
46. (Previously presented) The system of claim 30, wherein the queued component server is further adapted for removing a log event message from the server site event queue.
47. (Previously presented) The system of claim 46, wherein the queued component server is further adapted for removing the log event message in extensible markup language format from the server site event queue.
48. (Previously presented) The system of claim 47, wherein the queued component server is further adapted for removing the log event message from the server site event queue acting as an event processor.
49. (Previously presented) The system of claim 30, wherein the queued component server is further adapted for storing a log event message into the database.
50. (Previously presented) The system of claim 49, wherein the queued component server is further adapted for storing the log event message in extensible markup language format into the database.
51. (Previously presented) The system of claim 50, wherein the queued component server is further adapted for storing the log event message into a structured query language server data warehouse.

52. (Previously presented) The system of claim 51, further comprising means for analyzing the stored log event message.

53. (Previously presented) The system of claim 52, wherein the means for analyzing the stored log event message further comprises an online analytical processing application.

54. (Original) The system of claim 30, further comprising means for allowing a user to query the database via a web browser user interface.

55. (Original) The system of claim 54, wherein the means for allowing the user to query the database further comprises means for filtering query results based on selections entered by the user on the user interface.

56. (Original) The system of claim 55, wherein the means for filtering the query results further comprises means for displaying a report of the filtered results for the user via the user interface.

57. (Original) The method of claim 30, further comprising means for sending a notice of a security related event as an event notification to a predefined terminal for a system administrator.

58. (Previously presented) The method of claim 57, wherein the means for sending the notice of the security related event further comprises a filtering mechanism associated with the database for detecting the security event.